

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventors:	Richard Joseph Caballero, Mark David Lewis, Dita Naik		
Assignee:	Siebel Systems, Inc.		
Title:	SYSTEM AND METHOD FOR PROCESSING COMPLEX ORDERS		
Application No.:	10/024,691	Filing Date:	December 17, 2001
Examiner:	Siegfried E. Chencinski	Group Art Unit:	3691
Docket No.:	SBL0020US	Confirmation No.:	2964

---

Austin, Texas  
October 10, 2008

Mail Stop: Appeal Brief - Patents  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**APPEAL BRIEF**

Dear Sir:

This brief is submitted in support of the Notice of Appeal electronically filed on July 10, 2008 by the Appellants to the Board of Patent Appeals and Interferences from the Examiner's final rejection of claims 1-63. The Appellant notes that the appeal was received by the USPTO, thereby giving the appellant a period for filing set to expire on September 10, 2008. Filed herewith is a Petition for Extension of Time requesting a one-month extension, thereby giving the undersigned a period until October 10, 2008 in which to respond.

Please charge deposit account No. 502306 for the fee of \$510.00 associated with this Appeal Brief. Please charge this deposit account for any additional sums which may be required to be paid as part of this appeal.

REAL PARTY IN INTEREST

The assignee of the case, as named in the caption above, Siebel Systems, was acquired by Oracle International Corporation, which is therefore the real party in interest on this appeal.

RELATED APPEALS AND INTERFERENCES

There are no appeals or interferences related to this application.

STATUS OF CLAIMS

Claims 1-63 are pending in this application.

Claims 1-63 are rejected.

Appellant appeals the rejection of Claims 1-63.

STATUS OF AMENDMENTS

An amendment to Claims 1, 2, 4-24, and 27- 37 was filed on even date herewith. Appellants have respectfully requested entry of this amendment as it adds no new matter and simplifies the issues presented on appeal. These amendments are reflected in the Claim appendix attached hereto.

SUMMARY OF CLAIMED SUBJECT MATTER

Independent Claim 1 sets forth a complex order processing system. As is described *e.g.*, at page 80, line 11 of the present application, the system comprises a processor. As is described *e.g.*, at page 79, line 32 of the present application, the system comprises a computer-readable storage medium. As is illustrated, *e.g.*, at Figure 8, reference number 802 and described, *e.g.*, at page 27, line 14 and pages 43-64 of the present application, the system further comprises a plurality of business service methods embodied as instructions on the computer-readable storage medium. As is illustrated, *e.g.*, at Figure 9C, reference number 912 and described, *e.g.*, at page 29, line 9 of the

present application, the system also comprises user interface instructions on the computer-readable storage medium configured to cause the processor to generate a customer portal view.

As is illustrated, *e.g.*, at Figure 9C, reference number 916 and described, *e.g.*, at page 29, line 11 of the present application, the customer portal view comprises a list of products and services associated with an account. As is illustrated, *e.g.*, at Figure 9C, reference number 920 and described, *e.g.*, at page 29, line 11 of the present application, the customer portal view comprises a list of quotes associated with the account. As is illustrated, *e.g.*, at Figure 9C, reference number 916 and described, *e.g.*, at page 29, line 11 of the present application, the customer portal view comprises a list of orders associated with the account. As is illustrated, *e.g.*, at Figure 9C, reference number 930 and described, *e.g.*, at page 29, line 13 of the present application, the customer portal view comprises a plurality of user-selectable options. As is illustrated, *e.g.*, at Figure 9C, reference number 930 and described, *e.g.*, at page 29, line 13 of the present application, the user interface instructions are configured to receive input from a user, wherein the input comprises a selection of at least a selected option of the plurality of user-selectable options.

As is illustrated, *e.g.*, at Figure 10A, and described, *e.g.*, at pages 31-43 of the present application, the system comprises a plurality of workflow processes embodied as instructions on the computer-readable storage medium, wherein at least one of the plurality of workflow processes is invoked in response to the selection of the at least the option of the plurality of user-selectable options, and the at least one of the plurality of workflow processes is configured to invoke at least one of the plurality of business service methods to perform a function associated with the selected option.

Independent Claim 19 sets forth an order processing system. As is described *e.g.*, at page 79, line 32 of the present application, the system comprises a computer-readable storage medium. As is described *e.g.*, at page 80, line 11 of the present application, the system comprises a processor. As described, *e.g.*, at page 43, line 15 of the present application, the system comprises a complex object on the computer-readable storage medium comprising at least one of a service profile, quote information, and order

information pertaining to an account. As is illustrated, *e.g.*, at Figure 1, reference number 130 and described, *e.g.*, at page 8, line 32 of the present application, the system also comprises a data manager configured to cause said processor to receive at least a portion of the service profile from an external server via an information network, and generate a business object based on the portion of the service profile. As is illustrated, *e.g.*, at Figure 4A, reference number 410 and described, *e.g.*, at page 15, line 19 of the present application, the system further comprises a transformation engine, wherein the transformation engine is configured to cause the processor to generate at least a portion of the complex object, and the transformation engine is further configured to cause the processor to generate the at least a portion of the complex object using the business object.

Independent Claim 36 recites an apparatus for processing orders. As is described *e.g.*, at page 79, line 32 of the present application, the apparatus comprises a computer-readable storage medium. As is described *e.g.*, at page 80, line 11 of the present application, the apparatus comprises a processor. As is illustrated, *e.g.*, at Figure 8, reference number 802 and described, *e.g.*, at page 27, line 14 and pages 43-64 of the present application, the apparatus further comprises a plurality of business service methods, embodied as instructions on the computer-readable storage medium. As is illustrated, *e.g.*, at Figure 9C, reference number 912 and described, *e.g.*, at page 29, line 9 of the present application, the apparatus further comprises means for generating a customer portal.

As is illustrated, *e.g.*, at Figure 9C, reference number 916 and described, *e.g.*, at page 29, line 11 of the present application, the means for generating a customer portal comprises a list of products and services associated with an account. As is illustrated, *e.g.*, at Figure 9C, reference number 920 and described, *e.g.*, at page 29, line 11 of the present application, the means for generating a customer portal comprises a list of quotes associated with the account. As is illustrated, *e.g.*, at Figure 9C, reference number 916 and described, *e.g.*, at page 29, line 11 of the present application, the means for generating a customer portal comprises a list of orders associated with the account. As is illustrated, *e.g.*, at Figure 9C, reference number 930 and described, *e.g.*, at page 29, line

13 of the present application, the customer portal view comprises a plurality of user-selectable options. As is illustrated, *e.g.*, at Figure 9C, reference number 930 and described, *e.g.*, at page 29, line 13 of the present application, the customer portal is accessible by a user, wherein the user is one of a customer service representative, a customer, and a service provider for the customer.

As is illustrated, *e.g.*, at Figure 9C, reference number 930 and described, *e.g.*, at page 29, line 13 of the present application, the apparatus further comprises means for receiving input from the user. As is illustrated, *e.g.*, at Figure 10A, and described, *e.g.*, at pages 31-43 of the present application, the apparatus further comprises means for invoking a workflow process, wherein the means for invoking the workflow process comprises a means for invoking at least one of the plurality of business service methods and the at least one of the plurality of business service methods is configured to perform a function associated with a selected option from among the plurality of user-selectable options. As described, *e.g.*, at page 43, line 15 of the present application, the apparatus comprises means for receiving at least one of a service profile, quote information, and order information pertaining to an account from an external server. As is illustrated, *e.g.*, at Figure 1, reference number 120 and described, *e.g.*, at page 7, line 23-29 of the present application, the apparatus further comprises means for generating a business object, wherein means for generating uses the at least one of the service profile, the quote information, and the order information. As is illustrated, *e.g.*, at Figure 4A, reference number 410 and described, *e.g.*, at page 15, line 19 of the present application, the apparatus further comprises means for transforming at least a portion of the business object into a complex object.

Independent Claim 50 recites a method for processing orders. As is illustrated, *e.g.*, at Figure 9C, reference number 912 and described, *e.g.*, at page 29, line 9 of the present application, the method comprises generating a customer portal. As is illustrated, *e.g.*, at Figure 9C, reference number 916 and described, *e.g.*, at page 29, line 11 of the present application, the generating a customer portal comprises a list of products and services associated with an account. As is illustrated, *e.g.*, at Figure 9C, reference number 920 and described, *e.g.*, at page 29, line 11 of the present application,

the generating a customer portal comprises a list of quotes associated with the account. As is illustrated, *e.g.*, at Figure 9C, reference number 916 and described, *e.g.*, at page 29, line 11 of the present application, the generating a customer portal comprises a list of orders associated with the account.

As is illustrated, *e.g.*, at Figure 9C, reference number 930 and described, *e.g.*, at page 29, line 13 of the present application, the generating the customer portal comprises a plurality of user-selectable options. As is illustrated, *e.g.*, at Figure 9C, reference number 930 and described, *e.g.*, at page 29, line 13 of the present application, the generating the customer portal comprises presenting the customer portal such that the customer portal is accessible by a user, wherein the user is one of a customer service representative, a customer, and a service provider for the customer. As is illustrated, *e.g.*, at Figure 10A, and described, *e.g.*, at pages 31-43 of the present application, the method further comprises invoking a workflow process, wherein the workflow process invokes at least one business service method, and the at least one business service method performs a function associated with a selected option from among the plurality of user-selectable options.

As described, *e.g.*, at page 43, line 15 of the present application, the method further comprises receiving at least one of a service profile, quote information, and order information pertaining to an account. As is illustrated, *e.g.*, at Figure 1, reference number 120 and described, *e.g.*, at page 7, line 23-29 of the present application, the method further comprises generating a business object, wherein the generating is based on the at least one of the service profile, the quote information, and the order information. As is illustrated, *e.g.*, at Figure 4A, reference number 410 and described, *e.g.*, at page 15, line 19 of the present application, the method further comprises transforming at least a portion of the business object into a complex object.

GROUND OF REJECTION TO BE REVIEWED ON APPEAL

- I. Rejection of claims 1-63 under 35 U.S.C. §101 as being directed to non-statutory subject matter.
- II. Rejection of Claims 1-63 under 35 U.S.C. § 112, first paragraph, as failing to support the claimed invention by either a clearly asserted utility or a well-established utility.
- III. Rejection of Claims 1-63 under 35 U.S.C. § 112, second paragraph, as failing to set forth the subject matter which applicant regards as the invention.
- IV. Rejection of Claims 1-63 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.
- V. Rejection of Claims 1-63 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,587,838 B1 to Esposito, *et al.*, (*Esposito*), in view of U.S. Patent No. 5,870,717 to Wiecha, (*Wiecha*) and Official Notice.

ARGUMENT

- I. The rejection of claims 1-63 under 35 U.S.C. §101 as being directed to non-statutory subject matter is unfounded and should be overturned.

Claims 1-63 stand rejected under 35 U.S.C. § 101 on the basis of the allegation that the claimed invention is directed to non-statutory subject matter. Appellants respectfully traverse this rejection. While not conceding that the prior claims are directed to non-statutory subject matter, but instead to expedite prosecution, Appellants have chosen to amend Claims 1, 2, 4-24, and 27- 37 in an amendment submitted on the same date as this brief. These amendments are reflected in the listing of the claims submitted

in the appendix attached hereto. Appellants respectfully disagree with the Final Office Action's rejection and have reserved the right, for example, in a continuing application, to establish that the previous claims are not directed to non-statutory subject matter.

Regarding independent claims 1, 19, 36 and 50, the Final Office Action states that:

For a claim to be statutory under 35 U.S.C. § 101 the following condition must be met: ... In the claim, the practical application of an algorithm or idea results in a useful, concrete, tangible result.

According to the above guidelines, Applicant's [*sic*] claims are limited to the manipulation of abstract ideas in the context of patentability because the claim lacks an indication of usefulness.

*Final Office Action*, p. 2. Appellants interpret the position taken by the Final Office Action to be that of the claims somehow lacking usefulness by virtue of the claims reciting only the manipulation of abstract ideas. Appellants, respectfully, do not agree with any of these propositions.

Appellants respectfully submit that 35 U.S.C. § 101, relevant case law, and the Interim Guidelines for Examination of Patent Applications for Patent Subject Matter Eligibility of October 26, 2005, require nothing that is not already provided by the claims, as previously amended (or as originally submitted). As correctly noted in the Final Office Action, established case law makes clear the requirement of a "useful, concrete and tangible result." The appellants respectfully submit that claims 1-63 satisfy these requirements.

Regarding the need for a tangible result, the Interim Guidelines state:

The tangible requirement does not necessarily mean that a claim must either be tied to a particular machine or apparatus or must operate to change articles or materials to a different state or thing. However, the tangible requirement does require that the claim must recite more than a Sec. 101 judicial exception, in that the process claim must set forth a practical application of that Sec. 101 judicial exception to produce a real-world result. *Benson*, 409 U.S. at 71-72, 175 USPQ at 76-77 (invention



ineligible because had "no substantial practical application."). "[A]n application of a law of nature or mathematical formula to a . . . process may well be deserving of patent protection." Diehr, 450 U.S. at 187, 209 USPQ at 8 (emphasis added); see also Corning, 56 U.S. (15 How.) at 268, 14 L.Ed. 683 ("It is for the discovery or invention of some practical method or means of producing a beneficial result or effect, that a patent is granted . . ."). In other words, the opposite meaning of "tangible" is "abstract."

*Interim Guidelines*, p. 21. The operations recited in independent claims 1, 19, 36 and 50 satisfy this requirement in that they, among other things, provide for causing a processor to generate a customer portal view, which in turn allows a user to invoke a workflow process embodied as instructions on a computer-readable storage medium by selecting a user-selectable option. In so doing, the invocation of the workflow process causes the workflow process to invoke at least one of a plurality of business service methods, thereby causing a function associated with the selected option to be performed. Moreover, these operations are performed with specific structural requirements, both in terms of the physical apparatus (the processor) to which the invention is tied, and also in terms of structural requirements imposed on the software in use (the use of a user interface to allow a user to invoke a workflow process, and (in turn) a business service method, thereby causing a function associated with the selected option to be performed), a process that is significantly more structured and complex than simply allowing a user to select a function to perform. These operations do not provide an abstract result, but rather a tangible result, the invocation of a workflow process and, subsequently, a business service method, and so the performance of a function associated with the selected option. The customer portal view is also inherently tangible, as is made appearant in Figure 9C.

Regarding the need for a concrete result, the *Interim Guidelines* state:

Another consideration is whether the invention produces a "concrete" result. Usually, this question arises when a result cannot be assured. In other words, the process must have a result that can be substantially repeatable or the process must substantially produce the same result again. In *re Swartz*, 232 F.3d 862, 864, 56 USPQ2d 1703, 1704 (Fed. Cir. 2000) (where asserted result produced by the claimed invention is "irreproducible" claim should be rejected under section 101). The opposite of "concrete" is unrepeatable or unpredictable. Resolving this question is

dependent on the level of skill in the art. For example, if the claimed invention is for a process which requires a particular skill, to determine whether that process is substantially repeatable will necessarily require a determination of the level of skill of the ordinary artisan in that field. An appropriate rejection under 35 U.S.C. Sec. 101 should be accompanied by a lack of enablement rejection under 35 U.S.C. Sec. 112, paragraph 1, where the invention cannot operate as intended without undue experimentation. See *infra*.

*Interim Guidelines*, p. 22. The Final Office Action provides no rationale as to why any of claims 1-63 do not satisfy this requirement. Claims 1-63 recite operations that clearly produce a useful result (the invocation of a workflow process and, subsequently, a business service method, and so the performance of a function associated with the selected option), which is readily, substantially repeatable. Further, causing a processor to receive input from a user is repeatable. The practical, real-world useful result sought in the Final Office Action is therefore clear: the performance of a function associated with the selected option, via the invocation of a workflow process and, subsequently, a business service method, and so allow a user to process complex orders.

Appellants, however, have amended independent claims 1, 19, 36 and 50 to more clearly recite the relationships between the claimed elements that contribute to the aforementioned usefulness, as well as the useful, concrete, tangible results provided by the claimed invention. Thus, the amended independent claims now recite, for example, that the input received from the user comprises a selection of a least a selected option of the user-selectable options, and that at least one of the plurality of workflow processes is invoked in response to the selection of the at least the option of the plurality of user-selectable options. Claims 1, 19 and 36 recite a physical processor, and claims 1 and 19 expressly state that the instructions on the computer-readable storage medium are configured to cause the processor to undertake specific actions. These and other amendments more clearly demonstrate the relationships between the elements in question, as well as the fact that the claimed invention does indeed provide a “useful, concrete and tangible result.”

Further, amended Claims 1, 19 and 36 recite a computer-readable medium, which is undoubtedly tangible. Such a computer-readable medium is supported in the specification, by way of non-limiting example, at page 79, line 33. Similarly, amended

Claim 50 recites a presenting step. Presenting is supported, by way of non-limiting example of its tangible embodiments, at Figures 9B-9E and the supporting text at pages 29 (starting at line 9)-35.

Accordingly, the appellants respectfully submit that claims 1-63 satisfy the requirements of 35 U.S.C § 101.

The Examiner has courteously attempted to respond to this argument at page 7 of the Final Office Action, stating:

Applicant's claimed invention results in "transforming at least a portion of the business object into a complex object." (claim 50, last limitation).

Independent system claim 1 results in implementing the method through the last system limitation of "the at least one of the plurality of workflow processes is configured to invoke at least one of the plurality of business service methods to perform a function associated with the selected option."

Independent system claim 19 results in "the transformation engine is further configured to generate the at least a portion of the complex object using the business object."

Independent apparatus 36 results in a "means for transforming at least a portion of the business object into a complex object."

These end results fail to meet the requirements of 35 U.S.C. § 101 because they are mere software instructions which fail end in the required result. A qualifying method step result must result in a useful action which may be automated through a computer system and related apparatus, computer readable medium storing the software and related apparatus, not merely a software modification.

*Final Office Action*, p. 7. Appellants respectfully submit that the Final Office Action's response fails to recognize the several clearly tangible steps called specifically to attention. Among other things, the independent claims provide for the generation of a customer portal view, which in turn allows a user to invoke a workflow process by selecting a user-selectable option. This step is further made tangible by the amended claim's reliance upon a specific apparatus in the recited "causing said processor to"

limitations. This customer portal view is not addressed by the Final Office Action, and Appellants arguments as to the meaning of the cited sections of interim guidelines are not contested by the Final Office Action. Appellants respectfully submit that the Final Office Action has misapplied 35 U.S.C. § 101. For this reason, Appellants respectfully submit that the Final Office Action's rejection of Claims 1-63 is unfounded; Appellants respectfully request that the Final Office Action's rejection of Claims 1-63 under 35 U.S.C § 101 be overturned.

II. The rejection of Claims 1-63 under 35 U.S.C. § 112, first paragraph, as failing to support the claimed invention by either a clearly asserted utility or a well-established utility is unfounded and should be overturned.

Claims 1-63 stand rejected under 35 U.S.C. § 112, first paragraph, as allegedly failing to support the claimed invention by either a clearly asserted utility or a well-established utility. Appellants respectfully traverse this rejection. Appellants respectfully submit that, having demonstrated the utility of the claimed invention in responding to the rejection of claims 1-63 under 35 U.S.C § 101, Appellants have also demonstrated that the Specification provides information sufficient to allow one of skill in the art to practice and use the claimed invention.

The Examiner has courteously attempted to respond to this argument at page 9 of the Final Office Action, stating:

The MPEP states claims which fail to meet the requirements of 35 U.S.C. § 101 automatically fail to meet the requirements of 35 U.S.C. § 112-1<sup>st</sup> paragraph.

*Final Office Action*, p. 9. Appellants respectfully submit that the Appellants' arguments with respect to 35 U.S.C. § 101 have overcome the Final Office Action's rejection with respect to both 35 U.S.C. § 101 and the present rejection under 35 U.S.C. § 112, first paragraph. Appellants respectfully submit that the Final Office Action has misapplied

both 35 U.S.C. § 101 and 35 U.S.C. § 112, first paragraph. For this reason, Appellants respectfully submit that the Final Office Action's rejection of Claims 1-63 is unfounded; Appellants respectfully request that the Final Office Action's rejections of Claims 1-63 under 35 U.S.C § 112, first paragraph be overturned.

III. Rejection of Claims 1-63 under 35 U.S.C. § 112, second paragraph, as failing to set forth the subject matter which applicant regards as the invention.

Claims 1-63 further stand rejected under 35 U.S.C. § 112, second paragraph, as allegedly failing to set forth the subject matter which the applicant(s) regard as their invention. The Final Office Action posits the following:

“In that paper [the Specification filed December 17, 2001], applicant has stated the invention has the purpose of processing complex orders which include complex services and products, and this statement indicates that the invention is different from what is defined in the claim(s) because the independent claims do not include a limitation which indicates the processing of orders?????”

*Final Office Action*, p. 3. Appellants respectfully disagree. Each of the independent claims includes language that clearly demonstrates that the invention claimed thereby is concerned with processing complex orders which include complex services and products. For example, claim 1 is directed to a “complex order processing system” (which is affected by the claim limitations) and, at the very least explicitly recites display of a list of orders in a customer portal view. In claim 19, the claimed order processing system functions by managing and transforming complex objects representing at least one of services, orders and/or quotes. With regard to claim 36, an apparatus for processing orders is claimed, wherein business service methods are used to process orders for at least

one of services, orders and/or quotes, which are represented by business objects. Finally, claim 50 is directed to order processing functions by managing and transforming complex objects representing at least one of services, orders and/or quotes. As will be appreciated, in contrast to statements made in the Final Office Action, the independent claims, and so the dependent claims, are indeed directed to the processing of complex orders which include complex services and products.

The Examiner has courteously attempted to respond to this argument at page 9 of the Final Office Action, stating:

In spite of further modifying the claims, though only in minor ways, the applicant has failed to claim a method, system, apparatus and computer medium which accomplish the processing of orders. The clearly stated accomplishment is stated in the final limitation of claim 50, and is supported by the final limitations in the other independent claims of systems and apparatus with a computer readable medium, namely to “transforming at least a portion of the business object into a complex object” (claim 50, last limitation). This is why the claims continue to not meet the requirements of 35 U.S.C. 112-2<sup>nd</sup> paragraph.

*Final Office Action*, p. 9. The Examiner has concentrated on the final limitation of each claim. Appellants are aware of no precedent requiring such a connection between the literal wording of the preamble and the final limitation. Appellants respectfully submit that each of the independent claims, and likewise the dependent claims, relates to order-processing interactions in the explicit terms of at least one limitation that result in the requisite complex order processing. For this reason, Appellants respectfully request that the Examiner’s rejection of Claims 1-63 under 35 U.S.C § 112, second paragraph, be overturned.

- IV. The rejection of Claims 1-63 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention is unfounded and should be overturned.

Claims 1-63 further stand rejected under 35 U.S.C. § 112, second paragraph, as allegedly being indefinite for failing to particularly point out and distinctly claim the subject matter which the appellant regards as the invention. The Final Office Action posits:

For example, the limitation “at least one business service method performs a function associated with the selected option” in independent method claim 50 is vague and indefinite.” Independent Claims 1, 19 and 36 contain similar enabling limitations.

*See Final Office Action*, p. 4. Appellants have previously amended Claim 50 and Claim 36 to further clarify the indicated limitations, now reciting “a selected option from among the plurality of user-selectable options,” which Appellants believe renders moot the rejections of Claim 50 and Claim 36. Appellants have likewise amended Claim 19 to recite “the transformation engine is further configured to generate the at least a portion of the complex object using the business object.”

In the case of Claim 1, however, Appellants most respectfully disagree. The Final Office Action points to the limitation “the at least one of the plurality of workflow processes is configured to invoke at least one of the plurality of business service methods to perform a function associated with the selected option.” Appellants have reviewed Claim 1 and find that each of the three recited limitations of Claim 1 that is preceded by a definite article is also supported by an antecedent within Claim 1. Specifically, “the selected option”, which Appellants believe to be the basis of the Examiner’s rejection, is

supported with an antecedent “a selected option” at line 12 of Claim 1. For this reason, Appellants respectfully request that the Examiner’s rejection of Claims 1-63 under 35 U.S.C § 112, second paragraph, be overturned.

V. The rejection of Claims 1-63 under 35 U.S.C. § 103(a) as being unpatentable over *Esposito* in view of *Wiecha* and Official Notice is unfounded and should be overturned.

Claims 1-63 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Esposito* in view of *Wiecha* and Official Notice. While not conceding that the cited references qualify as prior art, but instead to expedite prosecution, Appellants have chosen to respectfully disagree and traverse the rejection as follows. Appellants have reserved the right, for example, in a continuing application, to establish that the cited references, or other references cited now or hereafter, do not qualify as prior art as to an invention embodiment previously, currently, or subsequently claimed.

In order for a claim to be rendered invalid under 35 U.S.C. § 103, the subject matter of the claim as a whole would have to be obvious to a person of ordinary skill in the art at the time the invention was made. *See* 35 U.S.C. § 103(a). This requires: (1) the references must teach or suggest all of the claim limitations; (2) there must be some teaching, suggestion or motivation to combine references either in the references themselves or in the knowledge of the art; and (3) there must be a reasonable expectation of success. *See* MPEP 2143; MPEP 2143.03; *In re Rouffet*, 149 F.3d 1350, 1355-56 (Fed. Cir. 1998).



Appellants have amended independent claims 1, 19, and 36 in the manner of claim 50, which now reads as follows:

50. A method for processing orders comprising:  
generating a customer portal, wherein  
the generating a customer portal comprises  
a list of products and services associated with an account,  
a list of quotes associated with the account,  
a list of orders associated with the account, and  
a plurality of user-selectable options, and  
presenting the customer portal such that the customer portal is  
accessible by a user, wherein  
the user is one of a customer service representative, a  
customer, and a service provider for the customer;  
invoking a workflow process, wherein  
the workflow process invokes at least one business service method,  
and  
the at least one business service method performs a function  
associated with a selected option from among the plurality  
of user-selectable options;  
receiving at least one of:  
a service profile,  
quote information, and  
order information pertaining to an account;  
generating a business object, wherein  
the generating is based on the at least one of  
the service profile,  
the quote information, and  
the order information; and  
transforming at least a portion of the business object into a complex  
object.

By contrast, *Esposito* is directed to a method and system for conducting real time electronic commerce, and describes:

... a system for providing real-time notification of purchaser requirements in a heterogeneous network environment .... The heterogeneous network environment can include a public packet switched network, such as the Internet, and a wireless network, such as a plurality of pagers. Select embodiments include a plurality of wireless terminals in communication with the wireless network to receive real-time notification of a purchaser's requirements. At least one vendor computer is part of the system. The vendor computer has one or more databases for storing

information about products and services in a product database, and a customer database, for storing information about customers. A plurality of client computers are part of the system. The client computers enable users to interact with the system to learn about and express interest in products and services.

*See Esposito*, Abstract. By further contrast, *Wiecha* is directed to a system for ordering items over computer network using an electronic catalog, and describes an approach in which:

Current corporate purchasing procedures are labor-intensive and therefore costly. The system enables an employee who needs an item which must be ordered from a supplier to select the item from an electronic catalog displayed on a personal computer and submit an order for approval and processing directly, by-passing both the normal paper approvals and the manual verification of the order by the organization's Purchasing department. It achieves this by means of an electronic catalog accessible from the employee's own personal computer, and a computer network and associated services linking the enterprise to one or more suppliers.

*See Wiecha*, Abstract.

As an initial matter, the Final Office Action states that, basically, the Abstract of *Esposito* teaches the claimed invention, save for construction of the software for processing complex customer inquiries and purchase orders. (*See* Final Office Action, p. 5) No specific citation, *e.g.*, page and line, to *Wiecha* is made at all with regard to independent claims 1, 19, 36 and 50. With regard to the dependent claims, no citation whatsoever is made to *Esposito* or *Wiecha*. Appellant is left to conclude that these claims are rejected simply for depending on a rejected claim. The Final Office Action merely picks a few of the claims, states that certain of such terminology such as that used in the claims is known (a point that Appellants do not concede), and then goes on to state that

such are obvious as such terminology is known to users of systems from SAP, Baan and Oracle (once again, a point that Appellants do not concede).

Appellants therefore respectfully submit that the particular parts of the cited references that are relied upon in the Final Office Action have not been designated as nearly as practicable, and the pertinence of each reference has not been clearly explained, both as required by 37 C.F.R. § 1.104(c)(2). *See also* MPEP § 706.02(j). Moreover, the Final Office Action then proceeds to summarily make a bare assertion of obviousness of almost every limitation of the independent claims. Again, no mention is made of the references that would enable Appellants to ascertain which sections of the references the Final Office Action purports to map to the limitations of Appellants' claims. Appellants respectfully request, in light of the present Request for Continuing Examination, that the Examiner be admonished to, in the next office action, more thoroughly discharge the obligation that the particular parts of the cited references that are relied upon in the office action be designated as nearly as practicable, and the pertinence of each reference be clearly explained.

Further, the rejection is based upon a list of various items, some within the references and some which are not, that are loosely, if at all, connected to various elements in Appellants' claims without further support. Further still, the items recited in the references (such as they are) are generally unrelated to one another, resulting in inconsistency, and clearly have very little to do with Appellants' claims as recited. Appellants further object to the Examiner's various assertions of "inherency" in various elements in that such inherencies are either false or otherwise insufficient to sustain the rejection. Nevertheless, Appellants have made every effort to respond to the rejections outlined in the Final Office Action.

However, the Final Office Action does correctly state that *Esposito* fails to teach the construction of the software for processing complex customer inquiries and purchase orders. (*Final Office Action*, p. 5) For example, nowhere in *Esposito* is there shown, taught or suggested the generation of a customer portal. Moreover, nowhere is there shown, taught or suggested the invocation of a workflow process and, subsequently, a business service method, and so the performance of a function associated with a selected option. This is the case because *Esposito* teaches a "...a system for providing real-time notification of purchaser requirements in a heterogeneous network environment ...." (*Esposito*, Abstract). Thus, *Esposito* is directed to conducting real time electronic commerce, and is not equipped to deal with complex order processing, as in the claimed invention.

*Wiecha* is directed to a system for ordering items over computer network using an electronic catalog. However, there is nothing taught by *Wiecha* that demonstrates anything more than simply a computer-based catalog system that allows a user to order items in the catalog electronically. This is just a system enables an employee who needs an item which must be ordered from a supplier to select the item from an electronic catalog displayed on a personal computer and submit an order for approval and processing directly, by-passing both the normal paper approvals and the manual verification of the order by the organization's Purchasing department.

By marked contrast, the claimed invention is directed, for example, to the processing of complex orders by the generation of a customer portal. Moreover, nowhere is there shown, taught or suggested the invocation of a workflow process and, subsequently, a business service method, and so the performance of a function associated with a selected option. Thus, neither *Esposito* nor *Wiecha*, taken alone or in permissible

combination, shows, teaches or suggests (or has reason to do so) the use of the claimed invocations to address a user's need to perform certain functions in providing processing of complex orders.

In addition to the foregoing infirmities, this is because neither *Esposito* nor *Wiecha*, taken separately or in any permissible combination, have need nor recognition of technology such as that provided by the claimed invention. That neither reference teaches such technology thus comes as no surprise. In *Esposito*, the chosen approach is preferred in order to provide real-time notification of purchaser requirements in a heterogeneous network environment, and so the use of the claimed technology goes unrecognized and unappreciated. Given that the claimed processing of complex orders is achieved through the invocation of a workflow process and, subsequently, a business service method, and so the performance of a function associated with a selected option, such functionality cannot be exercised over their design in systems employing the claimed invention. *Wiecha* fails to cure this infirmity of *Esposito*.

To wit, while *Esposito* may discuss a system which enables an employee who needs an item which must be ordered from a supplier to select the item from an electronic catalog displayed on a personal computer and submit an order for approval and processing directly, nothing is shown, taught or suggested with regard to the invocation of a workflow process and, subsequently, a business service method, and so the performance of a function associated with a selected option. This at least for the reason that *Wiecha* is concerned with avoiding the traditional use of paper documents and used for Purchasing department approval. Given that *Wiecha* has no need of the claimed technology in achieving this end, one of skill in the art would not look to *Wiecha*'s disclosure for such teachings. In fact, given that *Wiecha* fails to demonstrate any sort of

need for such technology, one would not expect one of skill in the art to look anywhere therefore. Thus, *Wiecha* provides neither a way to modify *Esposito*, nor does either reference demonstrate any sort of a need for such technology.

Thus, the Final Office Action does not establish the presence of these limitations in *Esposito* or *Wiecha*, taken alone or in permissible combination. As the Examiner appreciates, the burden is of supporting a case of obviousness lies with the Office, including whether the prior art references teach or suggest all of the claim limitations. See MPEP 706.02(j). For the reasons presented above, neither *Esposito* nor *Wiecha*, taken alone or in permissible combination, teach these limitations of claim 50, as well as the remaining independent claims.

In addition, Appellants also respectfully submit that the Final Office Action fails to satisfy the burden of factually supporting the alleged motivation to combine the two references. The duty of factually supporting the alleged motivation to combine may not be satisfied by engaging impermissible hindsight; any conclusion of obviousness must be reached on the basis of facts gleaned from the references. Evidence must therefore be provided to suggest the combination, and, as will be appreciated, “[b]road conclusory statements regarding the teaching of multiple references, standing alone, are not ‘evidence.’” See *In re Dembiczak*, 50 U.S.P.Q.2d 1614, 1617 (Fed. Cir. 1999). The Final Office Action’s failure to meet this criteria is particularly evident with regard to the dependent claims, wherein neither reference is cited as teaching any recited limitation, nor is Official Notice taken thereof (which, Appellants maintain, would be inappropriate in any event).

In this regard, Appellants respectfully submit that *Esposito* and *Wiecha*, taken alone or in permissible combination, are not capable of presenting a system in which the

invocation of a workflow process and, subsequently, a business service method, and so the performance of a function associated with a selected option. Such technology is simply not shown, taught or suggested by either of these references. For the claimed invention, the ability to process complex orders in the manner discussed in the specification and recited in the claims, the combination of the use of *Esposito* and *Wiecha*, particularly given that such is achieved through the invocation of a workflow process and, subsequently, a business service method, and so the performance of a function associated with a selected option.

This leads to the conclusion that not only do *Esposito* and *Wiecha* fail to show, teach or suggest such technology, but have no need to do so, because neither recognizes these benefits. Failing to appreciate these benefits, neither references addresses the obstacles that exist to creating solutions, nor providing any such solutions, such as those provided by the claimed invention. Lacking such appreciation and teachings, one of skill in the art could not find any motivation to combine their disclosures, because neither could be seen as lacking in this regard. Save for having the present disclosure to use as a blueprint, and so implying the use of hindsight, an obviously impermissible approach, which need not be dealt with here.

Further, the Final Office Action does not establish that such a combination of the teachings of these references would meet with success, as required. This comes as no surprise, as there is no way to combine these references to achieve the claimed invention, as demonstrated by the foregoing. Moreover, there is no teaching in the references, nor any extant in the skill in the art, which would provide the teachings necessary to successfully combine the disclosures of *Esposito* and *Wiecha*. Certainly, each of *Esposito*

and *Wiecha* are directed to their own solution to their own problems, and are self-sufficient and self-contained solutions in that regard.

Were one of skill in the art to proceed regardless (a fruitless effort and one which Appellants to not concede in any event), the combination would still fail to teach the claimed invention, if only in regard to the claimed invocation of a workflow process and, subsequently, a business service method, and so the performance of a function associated with a selected option. At best, such a combination would yield a system for ordering items over computer network using an electronic catalog, that would provide such services using real time electronic commerce. Thus, the claimed invention, by its recognition of the need for complex order processing, is in no way made obvious (nor anticipated) by *Esposito* and/or *Wiecha*, taken alone or in any permissible combination.

The Final Office Action responds to this argument by asserting that,

“In the instant case, the examiner has followed the Supreme Court’s upholding as cited above of the guidelines offered in the circuit Court’s *In re Kahn* opinion presented assessments of what the ordinary practitioner of the art would have known and would have derived by common sense at the time of Applicant’s invention. Applicant is referred to the prior art references and rationale presented in the above rejections. The examiner has not made “mere conclusory statements”, but has instead provided “rational underpinning to support the legal conclusion of obviousness”. As stated above, “the analysis need not seek out precise teachings directed to the specific subject matter of the challenged claim. Instead, the examiner has taken into account “the inferences and creative steps that a person of ordinary skill in the art would employ” by combining two prior art references with the stated rationale regarding what one of ordinary skill in the art would have known and found obvious at the time of Applicant’s invention. As such, the examiner considers that the above rejections meet the standards of a properly made *prima facie* case of obviousness rejection. The MPEP establishes a bar for Applicant to overcome in traversing the examiner’s prior art references and rejection rationale. The Examiner maintains the opinion that Applicant’s arguments fail to meet the requirements of that bar for a proper reversal.

See Final Office Action, p.13. Appellants respectfully disagree.



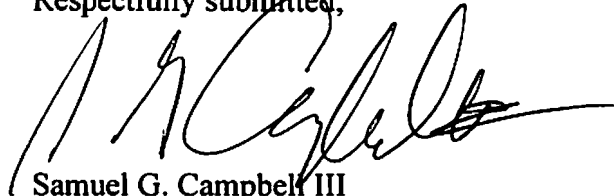
The Examiner correctly notes that the portion of *In re Kahn* cited by the *Ex Parte Catan* states that “Rejections based on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.” *See* Office Action, p.13. The Final Office Action’s failure to meet this criteria is particularly evident with regard to the dependent claims, wherein neither reference is cited as teaching any recited limitation, nor is Official Notice taken thereof (which, Appellants maintain, would be inappropriate in any event). Appellants respectfully submit that analysis implies the citation of evidence, supported by reasoning. The assertion that “the steps represent generically obvious steps of the well known order placement and acceptance process, mixed with special terminology...were well known to purchasing and customer service professional users... Therefore it would have been obvious...” is a naked assertion without adequate evidentiary support. Appellants respectfully submit that the Final Office Action has dismissed fifty-nine dependent claims without a single citation to any teaching of the two references on which the rejection is based and with only a passing mention of terms found in three of the fifty-nine dependent claims. Appellants respectfully submit that the dismissal of nearly thirty pages of claim language in less than one page of text from the Final Office Action is both conclusory and inadequate.

For these reasons, Appellants respectfully submit that the Final Office Action fails to present a *prima facie* case of obviousness of amended independent claims 1, 19, 36 and 50, and all claims dependent upon them, and that they are in condition for allowance. Appellants therefore respectfully submit that the Final Office Action’s rejection of Claims 1-63 under 35 U.S.C. §103 is unfounded and request that the rejections be reversed.

CONCLUSION

For the above reasons, Appellant respectfully submits that the rejection of pending Claims 1-63 is unfounded. Accordingly, Appellant respectfully requests that the Board reverse the rejections of these claims.

Respectfully submitted,

A handwritten signature in black ink, appearing to read 'S. G. Campbell III', written over a horizontal line.

Samuel G. Campbell III

Attorney for Appellants

Reg. No. 42,381

**CAMPBELL STEPHENSON LLP**

11401 Century Oaks Terrace

Building H, Suite 250

Austin, Texas 78758

Telephone: (512) 439-5084

Facsimile: (512) 439-5099

CLAIM APPENDIX

1. A complex order processing system comprising:
  - a processor;
  - a computer-readable storage medium;
  - a plurality of business service methods embodied as instructions on the computer-readable storage medium;
  - user interface instructions on the computer-readable storage medium configured to cause said processor to
    - generate a customer portal view comprising
      - a list of products and services associated with an account,
      - a list of quotes associated with the account,
      - a list of orders associated with the account, and
      - a plurality of user-selectable options, and
    - receive input from a user, wherein
      - the input comprises a selection of at least a selected option of the plurality of user-selectable options; and
  - a plurality of workflow processes embodied as instructions on the computer-readable storage medium, wherein
    - at least one of the plurality of workflow processes is invoked in response to the selection of the at least the option of the plurality of user-selectable options, and
    - the at least one of the plurality of workflow processes is configured to invoke at least one of the plurality of business service methods to perform a function associated with the selected option.

2. The complex order processing system of claim 1, further comprising a plurality of complex objects on the computer-readable storage medium, wherein each complex object is based on a property set of at least one of: a complex asset, a quote, and an order; and the business service methods include at least one of:

- an Apply Method to create a hybrid asset/order that simulates the future configuration of the complex asset by overlaying unprocessed items and attributes from open orders on one of the plurality of complex objects that is based on the complex asset;
- a Delta Method to generate a delta complex object that represents the difference between at least two of the plurality of complex objects;
- a Trim Method to create a new complex object by eliminating all items that meet predefined trim criteria from one of the plurality of complex objects;
- a Logical Delete Method to generate a transformed delta order complex object by transforming items in one of the plurality of complex objects with an action code “Delete” to “Update” and changing status of the transformed items to “Inactive”;
- an Explode Method to copy items in one of the plurality of complex objects with quantity greater than one, and generate a delta quote complex object with a corresponding number of copies of the item with quantity equal to one;
- a Fully Exploded Method to determine whether the items in one of the plurality of complex objects have a quantity greater than one;
- a Field Delta Method to set the action code of a parent quote item to “Update” when a specified field is updated;
- a Set Effectivity Dates Method to generate an “Effective End Date”, an “Effective Start Date”, and a “Completed Date”;
- a Set Action Method to set the action code of an item in one of the plurality of complex objects;
- a Save Object Method to receive a key and a property set hierarchy that represents one of the plurality of complex objects, and saves the one of the plurality of complex objects in a string to object map using the key;

- a Get Object Method to receive the key and return the one of the plurality of complex objects that corresponds to the key;
- a Reset Method to delete all saved complex objects from the map;
- a Module Licensed Method to determine whether the complex order processing system is licensed for a particular computer system;
- an Update Order Status Method to check a Status field of a root item and all child items in the hierarchy of one of the plurality of complex objects based on a complex asset to determine whether the Status field is set to “Complete”, “Failed” or NULL;
- a Merge Method to copy all items from one of the plurality of complex objects to another of the plurality of complex objects;
- a Set Field Method to enumerate through the item hierarchy of one of the plurality of complex objects, find a named field of each item, and set the named field to a specified value; and
- a Create Empty Complex Object Method to create an empty complex object based on one of the plurality of complex objects.

3. The complex order processing system of claim 2, wherein the workflow processes include at least one of:

- a New Products and Services Process to create a quote;
- a Modify Products and Services Process to modify a complex asset in a quote;
- a Disconnect Products and Services Process to disconnect a complex asset;
- a Profile Process to retrieve account information that corresponds to a quote;
- an Edit Quotes Line Items Process to apply all active and unprocessed orders to a complex asset to generate the future state of the complex asset, and allow the user to make changes to the configuration of the complex asset and update a quote;
- a Supplemental Orders Process to create a new order by allowing the user to revise a pending order;
- an Edit Order Line Items Process to apply all active and unprocessed orders to a complex asset to generate the future state of the complex asset, allow the

user to make changes to the configuration of the complex asset, and  
update an order;  
a Quote to Order Process to convert a quote to a sales order;  
an Update Opportunity Process to create revenue items for all quote line items;  
a Submit Order Process to associate account information with each line item in an  
order and set the status of the order to “open”;  
an Ungroup Quotes Process to create multiple copies of line items in a quote that  
have quantities greater than one;  
an Apply Completed Sales Order Line Item To Service Profile Process to update  
an asset table for a complex asset that has been provisioned; and  
a Move Process to transfer a customer’s complex asset configuration from an old  
location to a new location by entering the address of the new location and  
entering a single option selection.

4. The complex order processing system of claim 3, wherein the New  
Products and Services Process is configured to cause said processor to:

retrieve a selected quote;  
create a new quote if no quote is selected;  
designate the billing and service accounts; and  
display the active quote.

5. The complex order processing system of claim 3, wherein the Modify  
Products and Services Process is configured to cause said processor to:

retrieve the quote that is selected;  
create the quote if no quote is selected;  
allow the user to make changes to the currently requested state of the complex  
asset;  
apply changes specified by all open and pending orders related to the complex  
asset to simulate the state of the service product;  
return an empty asset when the open and pending order disconnects a root  
component of the complex asset;

allow the user to change the configuration of the complex asset when the complex asset is customizable;  
update the quote with the change between the previously requested state of the complex asset and the newly requested state of the complex asset; and  
store the updated quote.

6. The complex order processing system of claim 3, wherein the Disconnect Products and Services Process is configured to cause said processor to:

retrieve the quote that is selected;  
create the quote if no quote is selected; and  
allows the user to disconnect the complex asset, and create a quote line item with the complex asset disconnected.

7. The complex order processing system of claim 3, wherein the Edit Quotes Line Items Process is configured to cause said processor to:

determine whether the status of the action code of an order line item is "Add";  
generate the requested future state of the complex asset by applying the prior active and unprocessed orders to the complex asset;  
apply the current quote line item to the complex asset;  
display the currently requested state of the complex asset and allow the user to reconfigure the complex asset; and  
generate the difference between the requested state of the complex asset before the current order line item was applied and the state of the asset after the user reconfigured the complex asset.

8. The complex order processing system of claim 3, wherein the Edit Order Line Items Process is configured to cause said processor to:

determine whether the status of the action code of an order line item is "Add";  
generate the requested future state of the complex asset by applying the prior active and unprocessed orders to the complex asset;  
apply the current order line item to the complex asset;

display the currently requested state of the complex asset and allow the user to reconfigure the complex asset; and  
generate the difference between the requested state of the complex asset before the current order line item was applied and the state of the asset after the user reconfigured the complex asset; and  
ungroup any order line item with a quantity of more than 1 into multiple line items, each with a quantity of 1.

9. The complex order processing system of claim 3, wherein the Quote to Order Process is configured to cause said processor to:

ungroup any order line item that is tracked as an asset with a quantity of more than 1 into multiple line items, each with a quantity of 1  
create a new order based on the quote;  
set the order type of the new order to "Sales Order";  
set the order status of the new order to "New"; and  
set the status of each order line item "New" if the action field is non-null.

10. The complex order processing system of claim 3, wherein the Update Opportunity Process is configured to cause said processor to:

read the quote line items for a selected quote from the database.  
read the revenue items associated with any opportunity related to the selected quote;  
delete existing revenue items for the selected quote;  
create revenue items for each quote line item that has at least one of the following:  
an action code of Add or Update;  
a price type that is not Usage; and  
an extended quantity that is not zero.

11. The complex order processing system of claim 3, wherein the Submit Order Process is configured to cause said processor to:

determine whether the order status is not Open;



an order header with account information;  
propagate the account information to line items in the order;  
set the status of the order to Open; and  
set the status of pending line items to Open.

12. The complex order processing system of claim 3, wherein the Apply Completed Sales Order Line Item To Service Profile Process is configured to cause said processor to:

update an Order Item Processed flag of the topmost order line item based on whether all line items have a status of Complete.  
trim any incomplete or failed changes from the order line items whose products are not tracked as assets, line items whose status is not Complete, and line items whose action code is “-”;  
convert action codes of the order line items from Delete to Update;  
load the complex asset associated with the order line item from the database;  
create a new complex asset if there is no complex asset associated with the order line item;  
apply the order line item to the complex asset; and  
store the complex asset.

13. The complex order processing system of claim 3, wherein the Move Process is configured to cause said processor to:

identify a move-out address;  
identify a move-in address;  
retrieve the complex assets for the move-out address;  
generate a move-in quote with line items from the complex assets for the move-out address;  
add account information from the complex asset for the move-out address to the move-in quote;  
generate two quote line items for each quote line item in the move-out quote, wherein one quote line item has an action code of Delete and is associated

with the move-out address, and the other quote line item has an action code of Add and is associated with the move-in address.

specify a completed date for the move-out quote;

assign a service point to each quote line item to be connected in the move-in quote; and

specify a connect date for the move-in quote.

14. The complex order processing system of claim 2, wherein the Apply Method is configured to cause said processor to:

retrieve a complex asset representing a customizable product;

overlay unprocessed items and attributes of the customizable product from all of the open orders on the complex asset, wherein all of the open orders are represented in a property set in chronological order;

generate a new property set representing the combination of the complex asset and the open orders; and

set the hierarchy of the new property set to reflect the hierarchy of the open order.

15. The complex order processing system of claim 2, wherein the Delta Method is configured to cause said processor to:

determine when a change in the hierarchy occurs between a non-primary complex object and a primary complex object;

allow the user to indicate item fields and attributes in the two complex objects to be compared;

change the action code of an item to 'Add' when a new product is included in the non-primary complex object and not included in the primary complex object;

change the action code of an item to 'Delete' when a new product is included in the primary complex object and not included in the non-primary complex object;

change the action code of a child item to 'Update' when no other action is indicated;

change the action code of an item to 'Update' when the non-primary complex object's item and corresponding attribute(s) are added to the primary complex object; and

change the action code of the attribute of the item to:

'Update' when the attribute data has changed;

'Add' when the attribute has been added;

'Delete' when the attribute has been deleted; and

'-' (blank) when the attribute data has not changed,

when the action code of an item is changed to 'Update'.

16. The complex order processing system of claim 3, wherein the Trim Method is configured to cause said processor to:

receive a complex object, wherein the complex object represents a quotes, orders, and a service profile;

receive a trim specification, wherein the trim specification is a delta order begin with the top-most item in the complex object hierarchy, and proceeds recursively down through the root items to the trim any items and the children of the items that match criteria in the trim specification; and

allow syntax for the trim specification to support arithmetic functions and logical operators.

17. The complex order processing system of claim 3, wherein the Explode Method is configured to cause said processor to:

change the quantity of the line item if the action code on the line item is 'Add'; and

recognize a list of fields that should not be copied.

18. The complex order processing system of claim 3, wherein the Set Effectivity Dates Method is configured to cause said processor to:

generate the "Effective End Date" based on the date the action code of an item was set to "delete";

generate a “Calculated Effective Start Date” which is set to the date the action code of the item was set to “add”; and  
generate a “Completed Date” based on the date the order was completed.

19. An order processing system comprising:  
a computer-readable storage medium;  
a processor;  
a complex object on the computer-readable storage medium comprising at least one of  
a service profile,  
quote information, and  
order information pertaining to an account;  
a data manager configured to cause said processor to  
receive at least a portion of the service profile from an external server via  
an information network , and  
generate a business object based on the portion of the service profile; and  
a transformation engine, wherein  
the transformation engine is configured to cause said processor to generate  
at least a portion of the complex object, and  
the transformation engine is further configured to cause said processor to  
generate the at least a portion of the complex object using the  
business object.

20. The order processing system of claim 19, further comprising:  
user interface instructions on the computer-readable storage medium configured  
to cause said processor to:  
generate a customer portal view comprising a list of products and services  
associated with the account, a list of quotes associated with the  
account, a list of orders associated with the account, and a plurality  
of user-selectable options; and  
receive data input from a user.

21. The order processing system of claim 19, further comprising:  
a plurality of business service methods, embodied as instructions on the  
computer-readable storage medium, configured to cause said processor to  
perform a function associated with processing orders.
22. The order processing system of claim 21, further comprising:  
a plurality of workflow processes, embodied as instructions on the computer-  
readable storage medium, wherein each workflow process invokes at least  
one of the plurality of business service methods.
23. The order processing system of claim 20, further comprising:  
instructions on the computer-readable storage medium configured to cause said  
processor to update the status of order line items when a provisioning  
system completes all or part of an order.  
a plurality of business service methods;  
user interface instructions configured to cause said processor to:  
generate a customer portal view comprising a list of products and services  
associated with an account, a list of quotes associated with the  
account, a list of orders associated with the account, and a plurality  
of user-selectable options;  
receive input from a user; and  
a plurality of workflow processes, wherein at least one of the plurality of  
workflow processes is invoked when a user selects one of the plurality of  
user-selectable options, and the at least one of the plurality of workflow  
processes invokes at least one of the plurality of business service methods  
to perform a function associated with the selected option.
24. The complex order processing system of claim 23, further comprising a  
plurality of complex objects on the computer-readable storage medium, wherein each  
complex object is based on a property set of at least one of: a complex asset, a quote, and  
an order; and the business service methods include at least one of:

- an Apply Method to create a hybrid asset/order that simulates the future configuration of the complex asset by overlaying unprocessed items and attributes from open orders on one of the plurality of complex objects that is based on the complex asset;
- a Delta Method to generate a delta complex object that represents the difference between at least two of the plurality of complex objects;
- a Trim Method to create a new complex object by eliminating all items that meet predefined trim criteria from one of the plurality of complex objects;
- a Logical Delete Method to generate a transformed delta order complex object by transforming items in one of the plurality of complex objects with an action code "Delete" to "Update" and changing status of the transformed items to "Inactive";
- an Explode Method to copy items in one of the plurality of complex objects with quantity greater than one, and generate a delta quote complex object with a corresponding number of copies of the item with quantity equal to one;
- a Fully Exploded Method to determine whether the items in one of the plurality of complex objects have a quantity greater than one;
- a Field Delta Method to set the action code of a parent quote item to "Update" when a specified field is updated;
- a Set Effectivity Dates Method to generate a "Calculated Effective End Date", a "Calculated Effective Start Date", and a "Completed Date", wherein the "Calculated Effective End Date" is set to the date the action code of an item in one of the plurality of complex objects based on an order was set to "delete", the "Calculated Effective Start Date" is set to the date the action code of the item in the one of the plurality of complex objects was set to "add", and the "Completed Date" is set to the date the order was completed;
- a Set Action Method to set the action code of an item in one of the plurality of complex objects;

- a Save Object Method to receive a key and a property set hierarchy that represents one of the plurality of complex objects, and saves the one of the plurality of complex objects in a string to object map using the key;
- a Get Object Method to receive the key and return the one of the plurality of complex objects that corresponds to the key;
- a Reset Method to delete all saved complex objects from the map;
- a Module Licensed Method to determine whether the complex order processing system is licensed for a particular computer system;
- an Update Order Status Method to check a Status field of a root item and all child items in the hierarchy of one of the plurality of complex objects based on a complex asset to determine whether the Status field is set to “Complete”, “Failed” or NULL;
- a Merge Method to copy all items from one of the plurality of complex objects to another of the plurality of complex objects;
- a Set Field Method to enumerate through the item hierarchy of one of the plurality of complex objects, find a named field of each item, and set the named field to a specified value; and
- a Create Empty Complex Object Method to create an empty complex object based on one of the plurality of complex objects.

25. The complex order processing system of claim 21, wherein the workflow processes comprise:

- a Supplemental Orders Process to create a new order by allowing the user to revise a pending order, wherein the Supplemental Orders Process is configured to:
  - generate the requested future state of the complex object by applying the prior active and unprocessed orders to the complex object;
  - apply a current order line item to the complex object;
  - allow the user to reconfigure the complex object;

generate the difference between the requested state of the complex object before the current order line item was applied and the state of the object after the user reconfigured the complex object; and ungroup any order line item with a quantity of more than 1 into multiple line items, each with a quantity of 1.

26. The complex order processing system of claim 21, wherein the workflow processes comprise:

a Move Process to transfer a customer's complex asset configuration from an old location to a new location by entering the address of the new location and entering a single option selection to transfer the complex asset configuration from the old location to the new location.

27. The complex order processing system of claim 21, further comprising a plurality of complex objects on the computer-readable storage medium, wherein each complex object is based on a property set of at least one of: a complex asset, a quote, and an order; and the workflow processes comprise at least one of:

a New Products and Services Process to create a quote;  
a Modify Products and Services Process to modify a complex asset in a quote;  
a Disconnect Products and Services Process to disconnect a complex asset;  
a Profile Process to retrieve account information that corresponds to a quote;  
an Edit Quotes Line Items Process to apply all active and unprocessed orders to a complex asset to generate the future state of the complex asset, and allow the user to make changes to the configuration of the complex asset and update a quote;  
a Supplemental Orders Process to create a new order by allowing the user to revise a pending order;  
an Edit Order Line Items Process to apply all active and unprocessed orders to a complex asset to generate the future state of the complex asset, allow the user to make changes to the configuration of the complex asset, and update an order;



- a Quote to Order Process to convert a quote to a sales order;
- an Update Opportunity Process to create revenue items for all quote line items;
- a Submit Order Process to associate account information with each line item in an order and set the status of the order to “open”;
- an Ungroup Quotes Process to create multiple copies of line items in a quote that have quantities greater than one;
- an Apply Completed Sales Order Line Item To Service Profile Process to update an asset table for a complex asset that has been provisioned; and
- a Move Process to transfer a customer’s complex asset configuration from an old location to a new location by entering the address of the new location and entering a single option selection.

28. The complex order processing system of claim 27, wherein the Modify Products and Services Process is configured to cause said processor to:

- retrieve the quote that is selected;
- create the quote if no quote is selected;
- allow the user to make changes to the currently requested state of the complex asset;
- apply changes specified by all open and pending orders related to the complex asset to simulate the state of the service product;
- return an empty asset when the open and pending order disconnects a root component of the complex asset;
- allow the user to change the configuration of the complex asset when the complex asset is customizable; and
- update the quote with the change between the previously requested state of the complex asset and the newly requested state of the complex asset.

29. The complex order processing system of claim 27, wherein the Edit Quotes Line Items Process is configured to cause said processor to:

- determine whether the status of the action code of an order line item is “Add”;

generate the requested future state of the complex asset by applying the prior active and unprocessed orders to the complex asset;  
apply the current quote line item to the complex asset;  
display the currently requested state of the complex asset and allow the user to reconfigure the complex asset; and  
generate the difference between the requested state of the complex asset before the current order line item was applied and the state of the asset after the user reconfigured the complex asset.

30. The complex order processing system of claim 27, wherein the Edit Order Line Items Process is configured to cause said processor to:

determine whether the status of the action code of an order line item is "Add";  
generate the requested future state of the complex asset by applying the prior active and unprocessed orders to the complex asset;  
apply the current order line item to the complex asset;  
display the currently requested state of the complex asset and allow the user to reconfigure the complex asset; and  
generate the difference between the requested state of the complex asset before the current order line item was applied and the state of the asset after the user reconfigured the complex asset; and  
ungroup any line items in the complex asset with a quantity of more than 1 into multiple line items, each with a quantity of 1.

31. The complex order processing system of claim 27, wherein the Apply Completed Sales Order Line Item To Service Profile Process is configured to cause said processor to:

update an Order Item Processed flag of the topmost order line item based on whether all line items have a status of Complete.  
trim any incomplete or failed changes from the order line items whose products are not tracked as assets, line items whose status is not Complete, and line items whose action code is "-";

convert action codes of the order line items from Delete to Update;  
load the complex asset associated with the order line item from the database;  
create a new complex asset if there is no complex asset associated with the order  
line item; and  
apply the order line item to the complex asset.

32. The complex order processing system of claim 27, wherein the Move Process is configured to cause said processor to:  
identify a move-out address;  
identify a move-in address;  
retrieve the complex assets for the move-out address;  
generate a move-in quote;  
add account information from the complex asset for the move-out address to the  
move-in quote;  
generate two quote line items for each quote line item in the move-out quote,  
wherein one quote line item has an action code of Delete and is associated  
with the move-out address, and the other quote line item has an action  
code of Add and is associated with the move-in address;  
specify a completed date for the move-out quote; and  
assign a service point to each quote line item to be connected in the move-in  
quote.

33. The complex order processing system of claim 24, wherein the Apply Method is configured to cause said processor to:  
instructions to retrieve a complex asset representing a customizable product;  
instructions to overlay unprocessed items and attributes of the customizable  
product from all of the open orders on the complex asset, wherein all of  
the open orders are represented in a property set in chronological order;  
instructions to generate a new property set representing the combination of the  
complex asset and the open orders; and

instructions to set the hierarchy of the new property set to reflect the hierarchy of the open order.

34. The complex order processing system of claim 24, wherein the Delta Method is configured to cause said processor to:

- determine when a change in the hierarchy occurs between a non-primary complex object and a primary complex object;
- allow the user to indicate item fields and attributes in the two complex objects to be compared;
- change the action code of an item to 'Add' when a new product is included in the non-primary complex object and not included in the primary complex object;
- change the action code of an item to 'Delete' when a new product is included in the primary complex object and not included in the non-primary complex object;
- change the action code of a child item to 'Update' when no other action is indicated;
- change the action code of an item to 'Update' when the non-primary complex object's item and corresponding attribute(s) are added to the primary complex object; and
- change the action code of the attribute of the item to:
  - 'Update' when the attribute data has changed;
  - 'Add' when the attribute has been added;
  - 'Delete' when the attribute has been deleted; and
  - '-' (blank) when the attribute data has not changed,when the action code of an item is changed to 'Update'.

35. The complex order processing system of claim 24, wherein the Trim Method is configured to cause said processor to:

- receive a complex object, wherein the complex object represents a quotes, orders, and a service profile;

receive a trim specification, wherein the trim specification is a delta order begin with the top-most item in the complex object hierarchy, and proceeds recursively down through the root items to the trim any items and the children of the items that match criteria in the trim specification; and allow syntax for the trim specification to support arithmetic functions and logical operators.

36. An apparatus for processing orders comprising:  
a processor;  
a computer-readable storage medium;  
a plurality of business service methods, embodied as instructions on the computer-readable storage medium;  
means for generating a customer portal, wherein  
the means for generating a customer portal comprises  
a list of products and services associated with an account,  
a list of quotes associated with the account,  
a list of orders associated with the account, and  
a plurality of user-selectable options,  
the customer portal is accessible by a user, wherein  
the user is one of a customer service representative, a customer,  
and a service provider for the customer;  
means for receiving input from the user;  
means for invoking a workflow process, wherein  
the means for invoking the workflow process comprises a means for  
invoking at least one of the plurality of business service methods,  
and  
the at least one of the plurality of business service methods is configured  
to perform a function associated with a selected option from  
among the plurality of user-selectable options;  
means for receiving at least one of  
a service profile,

quote information, and  
order information pertaining to an account from an external server;  
means for generating a business object, wherein  
means for generating uses the at least one of  
the service profile,  
the quote information, and  
the order information; and  
means for transforming at least a portion of the business object into a complex  
object.

37. The apparatus of claim 36, further comprising:  
instructions on the computer-readable storage medium configured to cause said  
processor to update the status of order line items when a provisioning  
system completes all or part of an order.

38. The apparatus of claim 36, wherein the plurality of business service  
methods include at least one of:  
an Apply Method to create a hybrid asset/order that simulates the future  
configuration of the complex asset by overlaying unprocessed items and  
attributes from open orders on one of the plurality of complex objects that  
is based on the complex asset;  
a Delta Method to generate a delta complex object that represents the difference  
between at least two of the plurality of complex objects;  
a Trim Method to create a new complex object by eliminating all items that meet  
predefined trim criteria from one of the plurality of complex objects;  
a Logical Delete Method to generate a transformed delta order complex object by  
transforming items in one of the plurality of complex objects with an  
action code "Delete" to "Update" and changing status of the transformed  
items to "Inactive";

- an Explode Method to copy items in one of the plurality of complex objects with quantity greater than one, and generate a delta quote complex object with a corresponding number of copies of the item with quantity equal to one;
- a Fully Exploded Method to determine whether the items in one of the plurality of complex objects have a quantity greater than one;
- a Field Delta Method to set the action code of a parent quote item to “Update” when a specified field is updated;
- a Set Effectivity Dates Method to generate a “Calculated Effective End Date”, a “Calculated Effective Start Date”, and a “Completed Date”, wherein the “Calculated Effective End Date” is set to the date the action code of an item in one of the plurality of complex objects based on an order was set to “delete”, the “Calculated Effective Start Date” is set to the date the action code of the item in the one of the plurality of complex objects was set to “add”, and the “Completed Date” is set to the date the order was completed;
- a Set Action Method to set the action code of an item in one of the plurality of complex objects;
- a Save Object Method to receive a key and a property set hierarchy that represents one of the plurality of complex objects, and saves the one of the plurality of complex objects in a string to object map using the key;
- a Get Object Method to receive the key and return the one of the plurality of complex objects that corresponds to the key;
- a Reset Method to delete all saved complex objects from the map;
- a Module Licensed Method to determine whether the complex order processing system is licensed for a particular computer system;
- an Update Order Status Method to check a Status field of a root item and all child items in the hierarchy of one of the plurality of complex objects based on a complex asset to determine whether the Status field is set to “Complete”, “Failed” or NULL;
- a Merge Method to copy all items from one of the plurality of complex objects to another of the plurality of complex objects;

- a Set Field Method to enumerate through the item hierarchy of one of the plurality of complex objects, find a named field of each item, and set the named field to a specified value; and
- a Create Empty Complex Object Method to create an empty complex object based on one of the plurality of complex objects.

39. The apparatus of claim 36, wherein the workflow process comprises:

- a Supplemental Orders Process to create a new order by allowing the user to revise a pending order, wherein the Supplemental Orders Process comprises:

- means to generate the requested future state of the complex object by applying the prior active and unprocessed orders to the complex object;

- means to apply a current order line item to the complex object;

- means to allow the user to reconfigure the complex object; and

- means to generate the difference between the requested state of the complex object before the current order line item was applied and the state of the object after the user reconfigured the complex object; and

- means to ungroup any order line item with a quantity of more than 1 into multiple line items, each with a quantity of 1..

40. The apparatus of claim 36, wherein the workflow process comprises:

- a Move Process to transfer a customer's complex asset configuration from an old location to a new location by entering the address of the new location and entering a single option selection to transfer the complex asset configuration from the old location to the new location.

41. The apparatus of claim 36, wherein the workflow process comprises at least one of:

- a New Products and Services Process to create a quote;

- a Modify Products and Services Process to modify a complex asset in a quote;

- a Disconnect Products and Services Process to disconnect a complex asset;



a Profile Process to retrieve account information that corresponds to a quote;  
an Edit Quotes Line Items Process to apply all active and unprocessed orders to a complex asset to generate the future state of the complex asset, and allow the user to make changes to the configuration of the complex asset and update a quote;  
a Supplemental Orders Process to create a new order by allowing the user to revise a pending order;  
an Edit Order Line Items Process to apply all active and unprocessed orders to a complex asset to generate the future state of the complex asset, allow the user to make changes to the configuration of the complex asset, and update an order;  
a Quote to Order Process to convert a quote to a sales order;  
an Update Opportunity Process to create revenue items for all quote line items;  
a Submit Order Process to associate account information with each line item in an order and set the status of the order to “open”;  
an Ungroup Quotes Process to create multiple copies of line items in a quote that have quantities greater than one;  
an Apply Completed Sales Order Line Item To Service Profile Process to update an asset table for a complex asset that has been provisioned; and  
a Move Process to transfer a customer’s complex asset configuration from an old location to a new location by entering the address of the new location and entering a single option selection.

42. The apparatus of claim 41, wherein the Modify Products and Services Process comprises:  
means for retrieving the quote that is selected;  
means for creating the quote if no quote is selected;  
means for allowing the user to make changes to the currently requested state of the complex asset;  
means for applying changes specified by all open and pending orders related to the complex asset to simulate the state of the service product;

means for returning an empty asset when the open and pending order disconnects a root component of the complex asset;

means for allowing the user to change the configuration of the complex asset when the complex asset is customizable;

means for updating the quote with the change between the previously requested state of the complex asset and the newly requested state of the complex asset; and

means for storing the updated quote.

43. The apparatus of claim 41, wherein the Edit Quotes Line Items Process comprises:

means for determining whether the status of the action code of an order line item is “Add”;

means for generating the requested future state of the complex asset by applying the prior active and unprocessed orders to the complex asset;

means for applying the current quote line item to the complex asset;

means for allowing the user to reconfigure the complex asset; and

means for generating the difference between the requested state of the complex asset before the current order line item was applied and the state of the asset after the user reconfigured the complex asset.

44. The apparatus of claim 41, wherein the Edit Order Line Items Process comprises:

means for determining whether the status of the action code of an order line item is “Add”;

means for generating the requested future state of the complex asset by applying the prior active and unprocessed orders to the complex asset;

means for applying the current order line item to the complex asset;

means for displaying the currently requested state of the complex asset and allow the user to reconfigure the complex asset; and

means for generating the difference between the requested state of the complex asset before the current order line item was applied and the state of the asset after the user reconfigured the complex asset; and

means for ungrouping any order line item with a quantity of more than 1 into multiple line items, each with a quantity of 1.

45. The apparatus of claim 41, wherein the Apply Completed Sales Order Line Item To Service Profile Process comprises:

means for updating an Order Item Processed flag of the topmost order line item based on whether all line items have a status of Complete

means for trimming any incomplete or failed changes from the order line items whose products are not tracked as assets, line items whose status is not Complete, and line items whose action code is “-”;

means for converting action codes of the order line items from Delete to Update;

means for loading the complex asset associated with the order line item from the database;

means for creating a new complex asset if there is no complex asset associated with the order line item;

means for applying the order line item to the complex asset; and

means for storing the complex asset.

46. The apparatus of claim 41, wherein the Move Process comprises:

means for identifying a move-out address;

means for identifying a move-in address;

means for retrieving the complex assets for the move-out address;

means for generating a move-in quote with line items from the complex assets for the move-out address;

means for adding account information from the complex asset for the move-out address to the move-in quote;

means for generating two quote line items for each quote line item in the move-out quote, wherein one quote line item has an action code of Delete and is

associated with the move-out address, and the other quote line item has an action code of Add and is associated with the move-in address.

means for specifying a completed date for the move-out quote;

means for assigning a service point to each quote line item to be connected in the move-in quote; and

means for specifying a connect date for the move-in quote.

47. The apparatus of claim 38, wherein the Apply Method comprises:

means for retrieving a complex asset representing a customizable product;

means for overlaying unprocessed items and attributes of the customizable product from all of the open orders on the complex asset, wherein all of the open orders are represented in a property set in chronological order;

means for generating a new property set representing the combination of the complex asset and the open orders; and

means for setting the hierarchy of the new property set to reflect the hierarchy of the open order.

48. The apparatus of claim 38, wherein the Delta Method comprises:

means for determining when a change in the hierarchy occurs between a non-primary complex object and a primary complex object;

means for allowing the user to indicate item fields and attributes in the two complex objects to be compared;

means for changing the action code of an item to 'Add' when a new product is included in the non-primary complex object and not included in the primary complex object;

means for changing the action code of an item to 'Delete' when a new product is included in the primary complex object and not included in the non-primary complex object;

means for changing the action code of a child item to 'Update' when no other action is indicated;

means for changing the action code of an item to 'Update' when the non-primary complex object's item and corresponding attribute(s) are added to the primary complex object; and

means for changing the action code of the attribute of the item to:

'Update' when the attribute data has changed;

'Add' when the attribute has been added;

'Delete' when the attribute has been deleted; and

'-' (blank) when the attribute data has not changed,

when the action code of an item is changed to 'Update'.

49. The apparatus of claim 38, wherein the Trim Method comprises:

means for receiving a complex object, wherein the complex object represents a quotes, orders, and a service profile;

means for receiving a trim specification, wherein the trim specification is a delta order

means for beginning with the top-most item in the complex object hierarchy, and proceeds recursively down through the root items to the trim any items and the children of the items that match criteria in the trim specification; and

means for allowing syntax for the trim specification to support arithmetic functions and logical operators.

50. A method for processing orders comprising:

generating a customer portal, wherein

the generating a customer portal comprises

a list of products and services associated with an account,

a list of quotes associated with the account,

a list of orders associated with the account, and

a plurality of user-selectable options, and

presenting the customer portal such that the customer portal is accessible by a user, wherein

the user is one of a customer service representative, a customer,  
and a service provider for the customer;  
invoking a workflow process, wherein  
the workflow process invokes at least one business service method, and  
the at least one business service method performs a function associated  
with a selected option from among the plurality of user-selectable  
options;  
receiving at least one of:  
a service profile,  
quote information, and  
order information pertaining to an account;  
generating a business object, wherein  
the generating is based on the at least one of  
the service profile,  
the quote information, and  
the order information; and  
transforming at least a portion of the business object into a complex object.

51. The method of claim 50, further comprising:  
updating the status of order line items when a provisioning system completes all  
or part of an order.

52. The method of claim 50, wherein the plurality of business service methods  
include at least one of:

an Apply Method to create a hybrid asset/order that simulates the future  
configuration of the complex object by overlaying unprocessed items and  
attributes from open orders on one of the plurality of complex objects that  
is based on the complex object;  
a Delta Method to generate a delta complex object that represents the difference  
between at least two of the plurality of complex objects;

- a Trim Method to create a new complex object by eliminating all items that meet predefined trim criteria from one of the plurality of complex objects;
- a Logical Delete Method to generate a transformed delta order complex object by transforming items in one of the plurality of complex objects with an action code "Delete" to "Update" and changing status of the transformed items to "Inactive";
- an Explode Method to copy items in one of the plurality of complex objects with quantity greater than one, and generate a delta quote complex object with a corresponding number of copies of the item with quantity equal to one;
- a Fully Exploded Method to determine whether the items in one of the plurality of complex objects have a quantity greater than one;
- a Field Delta Method to set the action code of a parent quote item to "Update" when a specified field is updated;
- a Set Effectivity Dates Method to generate a "Calculated Effective End Date", a "Calculated Effective Start Date", and a "Completed Date", wherein the "Calculated Effective End Date" is set to the date the action code of an item in one of the plurality of complex objects based on an order was set to "delete", the "Calculated Effective Start Date" is set to the date the action code of the item in the one of the plurality of complex objects was set to "add", and the "Completed Date" is set to the date the order was completed;
- a Set Action Method to set the action code of an item in one of the plurality of complex objects;
- a Save Object Method to receive a key and a property set hierarchy that represents one of the plurality of complex objects, and saves the one of the plurality of complex objects in a string to object map using the key;
- a Get Object Method to receive the key and return the one of the plurality of complex objects that corresponds to the key;
- a Reset Method to delete all saved complex objects from the map;
- a Module Licensed Method to determine whether the complex order processing system is licensed for a particular computer system;

- an Update Order Status Method to check a Status field of a root item and all child items in the hierarchy of one of the plurality of complex objects based on a complex object to determine whether the Status field is set to “Complete”, “Failed” or NULL;
- a Merge Method to copy all items from one of the plurality of complex objects to another of the plurality of complex objects;
- a Set Field Method to enumerate through the item hierarchy of one of the plurality of complex objects, find a named field of each item, and set the named field to a specified value; and
- a Create Empty Complex Object Method to create an empty complex object based on one of the plurality of complex objects.

53. The method of claim 50, wherein the workflow process comprises:

- a Supplemental Orders Process to create a new order by allowing the user to revise a pending order, wherein the Supplemental Orders Process comprises:

- means to generate the requested future state of the complex object by applying the prior active and unprocessed orders to the complex object;

- means to apply a current order line item to the complex object;

- means to allow the user to reconfigure the complex object; and

- means to generate the difference between the requested state of the complex object before the current order line item was applied and the state of the object after the user reconfigured the complex object; and

- means to ungroup any order line item with a quantity of more than 1 into multiple line items, each with a quantity of 1..

54. The apparatus of claim 36, wherein the workflow process comprises:

- a Move Process to transfer a customer’s asset configuration from an old location to a new location by entering the address of the new location and entering a single option selection to transfer the asset configuration from the old location to the new location.



55. The method of claim 50, wherein the workflow process comprises at least one of:

- a New Products and Services Process to create a quote;
- a Modify Products and Services Process to modify a complex asset in a quote;
- a Disconnect Products and Services Process to disconnect a complex asset;
- a Profile Process to retrieve account information that corresponds to a quote;
- an Edit Quotes Line Items Process to apply all active and unprocessed orders to a complex asset to generate the future state of the complex asset, and allow the user to make changes to the configuration of the complex asset and update a quote;
- a Supplemental Orders Process to create a new order by allowing the user to revise a pending order;
- an Edit Order Line Items Process to apply all active and unprocessed orders to a complex asset to generate the future state of the complex asset, allow the user to make changes to the configuration of the complex asset, and update an order;
- a Quote to Order Process to convert a quote to a sales order;
- an Update Opportunity Process to create revenue items for all quote line items;
- a Submit Order Process to associate account information with each line item in an order and set the status of the order to “open”;
- an Ungroup Quotes Process to create multiple copies of line items in a quote that have quantities greater than one;
- an Apply Completed Sales Order Line Item To Service Profile Process to update an asset table for a complex asset that has been provisioned; and
- a Move Process to transfer a customer’s complex asset configuration from an old location to a new location by entering the address of the new location and entering a single option selection.

56. The method of claim 55, wherein the Modify Products and Services Process comprises:

- retrieving the quote that is selected;

creating the quote if no quote is selected;  
allowing the user to make changes to the currently requested state of the complex object;  
applying changes specified by all open and pending orders related to the complex object to simulate the state of the service product;  
returning an empty asset when the open and pending order disconnects a root component of the complex object;  
allowing the user to change the configuration of the complex object when the complex object is customizable; and  
updating the quote with the change between the previously requested state of the complex object and the newly requested state of the complex object.

57. The method of claim 55, wherein the Edit Quotes Line Items Process comprises:

determining whether the status of the action code of an order line item is “Add”;  
generating the requested future state of the complex object by applying the prior active and unprocessed orders to the complex object;  
applying the current quote line item to the complex object;  
allowing the user to reconfigure the complex object; and  
generating the difference between the requested state of the complex object before the current order line item was applied and the state of the asset after the user reconfigured the complex object.

58. The method of claim 55, wherein the Edit Order Line Items Process comprises:

determining whether the status of the action code of an order line item is “Add”;  
generating the requested future state of the complex object by applying the prior active and unprocessed orders to the complex object;  
applying the current order line item to the complex object;

displaying the currently requested state of the complex object and allow the user to reconfigure the complex object;  
generating the difference between the requested state of the complex object before the current order line item was applied and the state of the asset after the user reconfigured the complex object; and  
ungrouping any order line item with a quantity of more than 1 into multiple line items, each with a quantity of 1.

59. The method of claim 55, wherein the Apply Completed Sales Order Line Item To Service Profile Process comprises:

updating an Order Item Processed flag of the topmost order line item based on whether all line items have a status of Complete.  
trimming any incomplete or failed changes from the order line items whose products are not tracked as assets, line items whose status is not Complete, and line items whose action code is "-";  
converting action codes of the order line items from Delete to Update;  
loading the complex object associated with the order line item from the database;  
creating a new complex object if there is no complex object associated with the order line item;  
applying the order line item to the complex object; and  
storing the complex object.

60. The method of claim 55, wherein the Move Process comprises:

identifying a move-out address;  
identifying a move-in address;  
retrieving the complex objects for the move-out address;  
generating a move-in quote with line items from the complex objects for the move-out address;  
adding account information from the complex object for the move-out address to the move-in quote;

generating two quote line items for each quote line item in the move-out quote,  
wherein one quote line item has an action code of Delete and is associated  
with the move-out address, and the other quote line item has an action  
code of Add and is associated with the move-in address.

specifying a completed date for the move-out quote;

assigning a service point to each quote line item to be connected in the move-in  
quote; and

specifying a connect date for the move-in quote.

61. The method of claim 52, wherein the Apply Method comprises:  
retrieving a complex object representing a customizable product;  
overlaying unprocessed items and attributes of the customizable product from all  
of the open orders on the complex object, wherein all of the open orders  
are represented in a property set in chronological order;  
generating a new property set representing the combination of the complex object  
and the open orders; and  
setting the hierarchy of the new property set to reflect the hierarchy of the open  
order.

62. The method of claim 52, wherein the Delta Method comprises:  
determining when a change in the hierarchy occurs between a non-primary  
complex object and a primary complex object;  
allowing the user to indicate item fields and attributes in the two complex objects  
to be compared;  
changing the action code of an item to 'Add' when a new product is included in  
the non-primary complex object and not included in the primary complex  
object;  
changing the action code of an item to 'Delete' when a new product is included in  
the primary complex object and not included in the non-primary complex  
object;

changing the action code of a child item to 'Update' when no other action is indicated;

changing the action code of an item to 'Update' when the non-primary complex object's item and corresponding attribute(s) are added to the primary complex object; and

changing the action code of the attribute of the item to:

- 'Update' when the attribute data has changed;
- 'Add' when the attribute has been added;
- 'Delete' when the attribute has been deleted; and
- '-' (blank) when the attribute data has not changed,

when the action code of an item is changed to 'Update'.

63. The method of claim 52, wherein the Trim Method comprises:

receiving a complex object, wherein the complex object represents a quotes, orders, and a service profile;

receiving a trim specification, wherein the trim specification is a delta order beginning with the top-most item in the complex object hierarchy, and proceeds recursively down through the root items to the trim any items and the children of the items that match criteria in the trim specification; and

allowing syntax for the trim specification to support arithmetic functions and logical operators.

64-66. (Cancelled)

PATENT

EVIDENCE APPENDIX

None

RELATED PROCEEDINGS APPENDIX

None